Journal of Computational Physics 343 (2017) 414–431

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Direct numerical simulation of particulate flows with an overset grid method

A.R. Koblitz [a,*,3,4], S. Lovett [b], N. Nikiforakis [a], W.D. Henshaw [c,1,2]

[a] Department of Physics, Cavendish Laboratory, J J Thomson Avenue, Cambridge, CB3 0HE, UK
[b] Schlumberger Gould Research Centre, High Cross, Madingley Road, Cambridge, CB3 0EL, UK
[c] Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY 12180, USA

ARTICLE INFO

Article history:
Received 6 May 2016
Received in revised form 6 February 2017
Accepted 22 April 2017
Available online 2 May 2017

Keywords:
Particulate flow
Overset grids
Direct numerical simulation
Incompressible flow

ABSTRACT

We evaluate an efficient overset grid method for two-dimensional and three-dimensional particulate flows for small numbers of particles at finite Reynolds number. The rigid particles are discretised using moving overset grids overlaid on a Cartesian background grid. This allows for strongly-enforced boundary conditions and local grid refinement at particle surfaces, thereby accurately capturing the viscous boundary layer at modest computational cost. The incompressible Navier–Stokes equations are solved with a fractional-step scheme which is second-order-accurate in space and time, while the fluid–solid coupling is achieved with a partitioned approach including multiple sub-iterations to increase stability for light, rigid bodies. Through a series of benchmark studies we demonstrate the accuracy and efficiency of this approach compared to other boundary conformal and static grid methods in the literature. In particular, we find that fully resolving boundary layers at particle surfaces is crucial to obtain accurate solutions to many common test cases. With our approach we are able to compute accurate solutions using as little as one third the number of grid points as uniform grid computations in the literature. A detailed convergence study shows a 13-fold decrease in CPU time over a uniform grid test case whilst maintaining comparable solution accuracy.

1. Introduction

Flows of finite-sized particles in viscous fluids are common to many industrial as well as natural processes, such as primary cementing in the oil and gas industry [1] and blood flow [2]. Being so ubiquitous, particulate-flow problems span a large range of material and flow properties. Of interest to this work are flows of an incompressible Newtonian fluid with rigid, spherical (circular) particles of finite Reynolds number, where the Reynolds number describes the relative strength of inertial to viscous forces.

* Corresponding author.
E-mail addresses: ark44@cam.ac.uk (A.R. Koblitz), SLovett@slb.com (S. Lovett), nn10005@cam.ac.uk (N. Nikiforakis), henshw@rpi.edu (W.D. Henshaw).
[1] This work was supported by contracts from the U.S. Department of Energy ASCR Applied Math Program under grant AC52-07NA27344.
[2] Research supported by the National Science Foundation under grant DMS-1519934.
[3] Research supported by the Schlumberger Gould Research Centre under grant RG78221.
[4] This work was supported by the EPSRC Centre for Doctoral Training in Computational Methods for Materials Science under grant EP/L015552/1.

http://dx.doi.org/10.1016/j.jcp.2017.04.058
0021-9991/

Approximate solution methods have been applied to both high and low particle Reynolds number flow regimes, where by neglecting viscous or inertial contributions, respectively, the equations of motion can be linearised and solved with powerful mathematical tools; see [3–5] for examples in both flow regimes. It is the intermediate flow regime, where such approximations are not valid, that the full incompressible Navier–Stokes equations must be solved. A wide range of numerical techniques have been developed for simulating particulate flows through solution of the full Navier–Stokes equations. These include arbitrary Lagrangian–Eulerian (ALE) methods [6–8], methods based on level-sets [9,10], fictitious domain methods [11–13], embedded boundary methods [14] and immersed boundary methods (IBM) [15–27]. An extensive overview may be found in [28].

In this work we will evaluate the overset grid, or Chimera grid, method for viscous particulate flow. Overset grid methods have been widely used for problems with moving geometries. They were recognised early on to be a useful technique for treating rigid moving bodies, such as aircraft store separation [29], and have subsequently been applied to many other moving-grid aerodynamic applications, see for example [30–36]. English et al. [37] present a novel overset grid approach using a Voronoi grid to link Cartesian overset grids. This differs to the method used here, where interpolation stencils are directly substituted into the coefficient matrix and solid boundaries are represented using curvilinear grids. The basic approach of moving overset grids used in this article was developed for high-speed compressible and reactive flows by [31] and included the support for adaptive grid refinement. The deforming composite grid (DCG) approach was developed in [38] for treating deforming bodies with overset grids, and a partitioned scheme was developed for light deforming bodies that was stable without sub-iterations. A method to overcome the added-mass instability with compressible flows and rigid bodies was developed in [39]. More recently, stable partitioned schemes for incompressible flows and deforming solids have been developed [40–42] and extended to non-linear solids [43].

The method described in this paper retains much of the efficiency of static structured grid methods whilst still allowing for sharp representation of solid boundaries. The overset grid method can be seen as a bridge between the static grid methods such as IBM and boundary conformal grid methods; the curvilinear particle grids allow for higher than first-order accuracy and boundary conditions to be implemented strongly, while grid connectivity with the static Cartesian background grid is only locally updated. Since the grid connectivity is only updated locally, the regridding procedure is less costly and complex than for unstructured body conformal methods, such as ALE. Local grid refinement allows boundary layers to be fully resolved without appreciably affecting the total grid point count. This is in contrast with general static grid methods where the solver efficiency is offset by the unfavourable scaling associated with uniform grids, making large fully resolved simulations very costly [44]. For these reasons, we evaluate the suitability of the method for fully resolved simulations of incompressible fluid flow with rigid particles. Note that the scheme described here is implemented in the Cgins solver that is available as part of the Overture framework of codes (`overtureFramework.org`). Although past papers have described the use of Cgins for bodies undergoing specified motions (e.g. [45]), the discussion here of the algorithm involving freely-moving rigid-bodies is new.

In section 2 the overset grid method is summarised, before stating the mathematical formulation in section 3. These equations are discretised in space and time in section 4 and 5, respectively, while section 6 presents the fluid–solid coupling methodology. A grid convergence study is performed in section 7.1 using a representative test case to determine appropriate spatial resolutions for the wake structures captured by the background grid, and boundary layer captured by the particle grid. Following this, validation cases in both two and three space dimensions are presented, comparing against published experimental and numerical results. The results of our evaluation are summarised in section 8, along with an outlook towards future work.

2. Overset grids

The method of overset grids (also called overlapping, overlaid or Chimera grids) bridges the stationary and boundary conformal grid methods. A complex domain is represented by multiple body-fitted curvilinear grids that are allowed to overlap, as shown in Fig. 1. Overset grids bring flexibility to grid generation since component grids are not required to align along block boundaries. This flexibility allows component grids to be added in a relatively independent manner, requiring only local changes to grid connectivity.

A composite grid $\mathcal{G}$ consists of logically rectangular component grids $\mathcal{G}_k$, with $k = 1, 2, \ldots, N_g$. As illustrated in Fig. 1 the grid points of $\mathcal{G}$ are classified as interior points, boundary points, interpolation points and exterior or un-used points. The algorithm for generating a composite grid from a collection of component grids is intricate; a detailed description is beyond the remit of this work and so it will be only briefly discussed, with the intention of showing why the method of overset grids is appropriate for particulate flow problems. The interested reader is referred to [46] and [47] for a full description of the algorithm and implementation.

When moving grids are used, as is the case in particulate flow problems where each particle is represented by a separate component grid, the relative position of overset grids changes continuously. As a result, overlapping connectivity information, i.e. Chimera holes (regions of exterior points in the overset component grids) and interpolation points, must be recomputed at every time-step. Crucially, this is cheaper than complete grid regeneration and the required connectivity information recomputation can be locally confined to those grids affected by the moving grid.

Values of the solution at interpolation points are determined by standard tensor-product Lagrange-interpolation. We use quadratic interpolation (three-point stencil in each direction) for the results in this paper, as required for second-order ac-

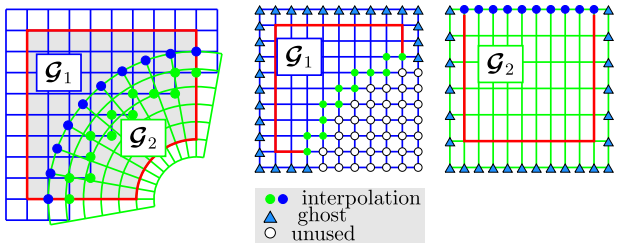

Fig. 1. Left: an overlapping grid consisting of two structured curvilinear component grids, $\boldsymbol{x} = \boldsymbol{\mathcal{G}}_1(\boldsymbol{r})$ and $\boldsymbol{x} = \boldsymbol{\mathcal{G}}_2(\boldsymbol{r})$. Middle and right: component grids for the square and annular grids in the unit square parameter space $\boldsymbol{r}$. Grid points are classified as discretisation points, interpolation points or unused points. Ghost points are used to apply boundary conditions.

curacy [46]. This interpolation is not locally conservative. Locally conservative interpolation on overset grids is possible [48] but has not been found necessary in our experience. Corrections to ensure global conservation are also possible and have been shown to have advantages, see for example [49].

3. Governing equations

Incompressible flow is governed by the Navier–Stokes equations,

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} = -\frac{1}{\rho}\nabla p + \nu\nabla^2\boldsymbol{u} + \boldsymbol{f},$$
$$\nabla\cdot\boldsymbol{u} = 0,$$

where $\boldsymbol{u}$ is the vector of Cartesian components of the velocity u_i, p the pressure field, ρ the fluid density, and $\nu = \mu/\rho$ the kinematic viscosity. For discretising the equations on a moving grid (on an overset grid some grids are static while others are attached to, and move with the body), we make a change of dependent variables $\boldsymbol{x}$ and t to a frame that moves with the grid. As a result, on moving domains, the governing equations transform to

$$\frac{\partial \boldsymbol{u}}{\partial t} + ((\boldsymbol{u}-\boldsymbol{w}))\cdot\nabla\boldsymbol{u} = -\frac{1}{\rho}\nabla p + \nu\nabla^2\boldsymbol{u} + \boldsymbol{f}, \tag{1}$$
$$\nabla\cdot\boldsymbol{u} = 0, \tag{2}$$

where $\boldsymbol{w}$ is the velocity of a point attached to the moving domain. The partial derivative in time in the moving frame, as appearing in (1), is therefore the derivative in time when keeping the spatial location fixed to a point that is attached to the moving domain. We solve an alternative formulation of system (1)–(2). A pressure Poisson equation is derived by taking the divergence of the momentum equation (1) and using (2). The resulting velocity–pressure formulation of the initial-boundary value problem is

$$\frac{\partial \boldsymbol{u}}{\partial t} + ((\boldsymbol{u}-\boldsymbol{w})\cdot\nabla)\boldsymbol{u} + \frac{1}{\rho}\nabla p - \nu\nabla^2\boldsymbol{u} - \boldsymbol{f} = 0, \qquad \forall \boldsymbol{x}\in\Omega, \tag{3}$$
$$\mathcal{J}(\nabla\boldsymbol{u}) + \frac{1}{\rho}\nabla^2 p - \nabla\cdot\boldsymbol{f} = 0, \qquad \forall \boldsymbol{x}\in\Omega, \tag{4}$$
$$B(\boldsymbol{u}, p) = \boldsymbol{0}, \qquad \forall \boldsymbol{x}\in\partial\Omega, \tag{5}$$
$$\nabla\cdot\boldsymbol{u} = 0, \qquad \forall \boldsymbol{x}\in\partial\Omega, \tag{6}$$
$$\boldsymbol{u}(\boldsymbol{x}, 0) = \boldsymbol{u}_0(\boldsymbol{x}), \qquad \text{at} \quad t = 0, \tag{7}$$

where $\mathcal{J}(\nabla\boldsymbol{u}) \equiv \nabla\boldsymbol{u} : \nabla\boldsymbol{u}$ and Ω denotes the fluid domain in n_d space dimensions. There are n_d primary boundary conditions, denoted by $B(\boldsymbol{u}, p) = \boldsymbol{0}$. The velocity–pressure formulation requires an additional boundary condition for the pressure. Here, the velocity divergence (6) is applied as the boundary condition on the pressure, making the velocity–pressure formulation equivalent to the velocity–divergence formulation [50]. For the second-order accurate scheme used here, boundary conditions are required to determine $\boldsymbol{u}$ and p at a line of fictitious (ghost) points outside the domain boundary. Some of the numerical boundary conditions are *compatibility conditions*, derived by applying the momentum and pressure equations on the boundary.

The motion of a rigid body immersed in the fluid is governed by the Newton–Euler equations,

$$\frac{\mathrm{d}\boldsymbol{x}_b}{\mathrm{d}t} = \boldsymbol{v}_b, \quad m_b\frac{\mathrm{d}\boldsymbol{v}_b}{\mathrm{d}t} = \boldsymbol{F}, \quad A\frac{\mathrm{d}\boldsymbol{\omega}}{\mathrm{d}t} = -\boldsymbol{\omega}\times A\boldsymbol{\omega} + \boldsymbol{T}, \quad \frac{\mathrm{d}\boldsymbol{e}_i}{\mathrm{d}t} = \boldsymbol{\omega}\times\boldsymbol{e}_i.$$

Here $\boldsymbol{x}_b(t)$ and $\boldsymbol{v}_b(t)$ are the position and velocity of the centre of mass, respectively, m_b is the mass of the body, $\boldsymbol{\omega}$ is the angular velocity, A is the moment of inertia matrix, $\boldsymbol{e}_i$ are the principal axes of inertia, $\boldsymbol{F}(t)$ is the applied force, and $\boldsymbol{T}(t)$

is the applied torque about the centre of mass of the body. The principal axes of inertia are integrated over time to find the rotation matrix which is used to update positions, velocities and acceleration of points attached to the body surface.

The force and torque on the body are determined from both body forces, such as gravity, and hydrodynamic forces arising from the stresses exerted by the fluid on the body surface, Γ,

$$\boldsymbol{F} = \int_\Gamma (-p\boldsymbol{n} + \boldsymbol{\tau}\cdot\boldsymbol{n})\,\mathrm{d}s + \boldsymbol{f}_b, \qquad \boldsymbol{T} = \int_\Gamma (\boldsymbol{x}-\boldsymbol{x}_b)\times(-p\boldsymbol{n} + \boldsymbol{\tau}\cdot\boldsymbol{n})\,\mathrm{d}s + \boldsymbol{t}_b,$$

where $\boldsymbol{x}$ is a point on Γ, $\boldsymbol{\tau} = \mu(\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T)$ is the viscous stress tensor, $\boldsymbol{n}$ is the unit normal vector to the body surface (outward pointing from the fluid domain), $\boldsymbol{f}_b$ is any external body force and $\boldsymbol{t}_b$ is any external body torque.

4. Spatial discretisation

The equations of the velocity–pressure formulation are discretised to second-order accuracy in space using finite difference methods on overset grids, see [50] and [51]. An overset grid consists of logically rectangular grids that cover a grid region and overlap where they coincide. The solutions between adjacent grids are connected via interpolation conditions. Each component grid (numbered $k = 1, 2, \ldots, N_g$) is associated with a transformation $\boldsymbol{d}_k : \mathbb{R}^3 \to \mathbb{R}^3$ from the unit square, with coordinates denoted by $\boldsymbol{r} = (r_1, r_2, r_3)$, into physical space, $\boldsymbol{x} = (x_1, x_2, x_3)$, and denoted by $\boldsymbol{d}_k(\boldsymbol{r}, t) = \boldsymbol{x}(\boldsymbol{r}, t)$, which allows for body fitted grids of non-rectangular shapes. Consider solving the equations in three space dimensions on a square component grid $\mathcal{G}_k$, with grid spacing $h_m = 1/N_m$, for a positive integer N_m:

$$\mathcal{G}_k = \{\boldsymbol{x}_{\boldsymbol{i},k} \mid \boldsymbol{i} = (i_1, i_2, i_3),\ N_{m,a,k} - 1 \le i_m \le N_{m,b,k} + 1,\ m = 1, 2, 3\},$$

where $\boldsymbol{i} = (i_1, i_2, i_3)$ is a multi-index and a and b denote the beginning and end grid line numbers, respectively. Ghost points are included at the boundaries, $i_m = N_{m,a,k}$ or $i_m = N_{m,b,k}$, to facilitate discretising to second order. The component grid number k will be dropped in the following discussion. Let $\boldsymbol{U}_{\boldsymbol{i}} \approx \boldsymbol{u}(\boldsymbol{x}_{\boldsymbol{i}}, t)$, $\boldsymbol{W}_{\boldsymbol{i}} \approx \boldsymbol{w}(\boldsymbol{x}_{\boldsymbol{i}}, t)$, and $P_{\boldsymbol{i}} \approx p(\boldsymbol{x}_{\boldsymbol{i}}, t)$ be the numerical approximations to $\boldsymbol{u}$, $\boldsymbol{w}$ and p, respectively. The momentum and pressure equations are discretised with second-order finite difference stencils. The derivatives with respect to $\boldsymbol{r}$ are standard second-order centred finite difference approximations, for example,

$$\frac{\partial \boldsymbol{u}}{\partial r_m} \approx D_{r_m}\boldsymbol{U}_{\boldsymbol{i}} := \frac{\boldsymbol{U}_{\boldsymbol{i}+\boldsymbol{e}_m} - \boldsymbol{U}_{\boldsymbol{i}-\boldsymbol{e}_m}}{2h_m}, \qquad \frac{\partial^2 \boldsymbol{u}}{\partial r_m^2} \approx D_{r_m r_m}\boldsymbol{U}_{\boldsymbol{i}} := \frac{\boldsymbol{U}_{\boldsymbol{i}+\boldsymbol{e}_m} - 2\boldsymbol{U}_{\boldsymbol{i}-\boldsymbol{e}_m} + \boldsymbol{U}_{\boldsymbol{i}-\boldsymbol{e}_m}}{h_m^2},$$

where $\boldsymbol{e}_m$ is the unit vector in the m-th coordinate direction. Using the chain rule the derivatives with respect to $\boldsymbol{x}$ are defined as

$$\begin{aligned}
\frac{\partial \boldsymbol{u}}{\partial x_m} &= \sum_n \frac{\partial r_n}{\partial x_m}\frac{\partial \boldsymbol{u}}{\partial r_n} \approx D_{x_m}\boldsymbol{U}_{\boldsymbol{i}} := \sum_n \frac{\partial r_n}{\partial x_m} D_{r_n}\boldsymbol{U}_{\boldsymbol{i}},\\
\frac{\partial^2 \boldsymbol{u}}{\partial x_m^2} &= \sum_{n,l} \frac{\partial r_n}{\partial x_m}\frac{\partial r_l}{\partial x_m}\frac{\partial^2 \boldsymbol{u}}{\partial r_n \partial r_l} + \sum \frac{\partial^2 r_n}{\partial x_m^2}\frac{\partial \boldsymbol{u}}{\partial r_n}\\
&\approx D_{x_m}D_{x_m}\boldsymbol{U}_{\boldsymbol{i}} := \sum_{n,l} \frac{\partial r_n}{\partial x_m}\frac{\partial r_l}{\partial x_m} D_{r_m r_l}\boldsymbol{U}_{\boldsymbol{i}} + \sum_n \left(D_{x_m}\frac{\partial r_n}{\partial x_m}\right) D_{r_n}\boldsymbol{U}_{\boldsymbol{i}},
\end{aligned}$$

where the entries in the Jacobian matrix, $\partial r_m/\partial x_n$ are obtained from the mapping $\boldsymbol{x} = \boldsymbol{d}_k(\boldsymbol{r}, t)$.

The discretised governing equations are

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{U}_{\boldsymbol{i}} + ((\boldsymbol{U}_{\boldsymbol{i}} - \boldsymbol{W}_{\boldsymbol{i}})\cdot\nabla_2)\boldsymbol{U}_{\boldsymbol{i}} + \frac{1}{\rho}\nabla_2 P_{\boldsymbol{i}} - \nu\nabla_2^2\boldsymbol{U}_{\boldsymbol{i}} - \boldsymbol{f}_{\boldsymbol{i}} &= 0,\\
\frac{1}{\rho}\nabla_2^2 P_{\boldsymbol{i}} + \mathcal{J}(\nabla_2\boldsymbol{U}_{\boldsymbol{i}}) - \nabla_2\cdot\boldsymbol{f}_{\boldsymbol{i}} &= 0,
\end{aligned}$$

where $\nabla_2\boldsymbol{U}_{\boldsymbol{i}} = (D_{x_1}\boldsymbol{U}_{\boldsymbol{i}}, D_{x_2}\boldsymbol{U}_{\boldsymbol{i}}, D_{x_3}\boldsymbol{U}_{\boldsymbol{i}})$, $\nabla_2^2\boldsymbol{U}_{\boldsymbol{i}} = (D_{x_1x_1} + D_{x_2x_2} + D_{x_3x_3})\boldsymbol{U}_{\boldsymbol{i}}$, and $\nabla_2\cdot\boldsymbol{U}_{\boldsymbol{i}} = D_{x_1}\boldsymbol{U}_{1,\boldsymbol{i}} + D_{x_2}\boldsymbol{U}_{2,\boldsymbol{i}} + D_{x_3}\boldsymbol{U}_{3,\boldsymbol{i}}$.

5. Temporal discretisation

The method of lines is used for solving the equations in time. After discretising the governing equations in space they can be regarded as a system of ODEs,

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{U} = \boldsymbol{F}(\boldsymbol{U}, \boldsymbol{t}),$$

where pressure is considered a function of the velocity, $P = P(\boldsymbol{U})$. The equations are integrated in time using either a fully explicit or semi-implicit scheme, depending on the stability restriction imposed by the viscous time-step characteristic

to the problem. The explicit scheme uses a second-order accurate Adams–Bashforth predictor followed by a second-order accurate Adams–Moulton corrector. For light rigid bodies, multiple correction steps are used to stabilise the scheme, under-relaxing the computed forces on the bodies. The semi-implicit scheme treats the viscous term of the momentum equation implicitly with a second-order Crank–Nicolson method, which is once again combined with Adams–Moulton corrector steps if under-relaxed sub-iterations are required. To illustrate this we will use the momentum equations as an example. Splitting the equations into explicit and implicit components we have

$$\frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}t} = -((\boldsymbol{u}-\boldsymbol{w})\cdot\nabla)\boldsymbol{u} - \frac{1}{\rho}\nabla p \;\; + \nu\nabla^2\boldsymbol{u} \;\; \equiv \boldsymbol{F}_E + \boldsymbol{F}_I,$$

where $\boldsymbol{F}_E$ and $\boldsymbol{F}_I$ are the explicit and implicit components, respectively

$$\boldsymbol{F}_E = -((\boldsymbol{u}-\boldsymbol{w})\cdot\nabla)\boldsymbol{u} - \frac{1}{\rho}\nabla p, \quad \boldsymbol{F}_I = \nu\nabla^2\boldsymbol{u}. \tag{8}$$

The equations are integrated using either fully explicit or semi-implicit schemes. The explicit integration scheme used in the present work is the second-order in time Adams predictor–corrector method. It consists of an Adams–Bashforth predictor

$$\frac{\boldsymbol{u}^p - \boldsymbol{u}^n}{\Delta t} = \beta_0\boldsymbol{F}^n + \beta_1\boldsymbol{F}^{n-1},$$

with the constants $\beta_0 = 1 + \frac{\Delta t}{2\Delta t_1}$ and $\beta_1 = -\frac{\Delta t}{2\Delta t_1}$ chosen for second-order accuracy even with a variable time-step where $\Delta t_1 = t_n - t_{n-1}$, and an Adams–Moulton corrector

$$\frac{\boldsymbol{u}^{n+1} - \boldsymbol{u}^n}{\Delta t} = \frac{1}{2}\boldsymbol{F}^p + \frac{1}{2}\boldsymbol{F}^n.$$

Though only one corrector step was taken here, one may in practice correct multiple times. In fact, this is necessary when dealing with moving, light rigid bodies and partitioned fluid–solid coupling, as will be discussed later.

A semi-implicit approach is taken in some diffusion dominated problems where the explicit diffusive time-step is overly restrictive. Generally, this is the case when the Reynolds number is very low or the grid is highly refined near solid boundaries. Here, the non-linear convective terms are treated with the explicit Adams predictor–corrector method while the viscous terms are treated with the implicit second-order in time Crank–Nicolson method. Using the notation introduced in (8) then the time-step consists of a predictor,

$$\frac{\boldsymbol{u}^p - \boldsymbol{u}^n}{\Delta t} = \beta_0\boldsymbol{F}_E^n - \beta_1\boldsymbol{F}_E^{n-1} + \alpha\boldsymbol{F}_I^p + (1-\alpha)\boldsymbol{F}_I^n,$$

and a corrector

$$\frac{\boldsymbol{u}^c - \boldsymbol{u}^n}{\Delta t} = \frac{1}{2}\boldsymbol{F}_E^p + \frac{1}{2}\boldsymbol{F}_E^n + \alpha\boldsymbol{F}_I^c + (1-\alpha)\boldsymbol{F}_I^n,$$

where the superscript c denotes the corrected solution and $\alpha = \frac{1}{2}$ gives the second-order Crank–Nicolson method.

The basic Navier–Stokes solver uses a solution algorithm that decouples the pressure and velocity fields [50,51] in a similar fashion to many fractional-step and projection schemes, cf. [52–54] and many others. The advantage of the current scheme over typical projection schemes is that the boundary conditions for the pressure are well-defined and it is straightforward to obtain full second-order accuracy for all variables.

Assume that at time $t - \Delta t$ the values of $\boldsymbol{U}(t-\Delta t)$ and $P(t-\Delta t)$ are known at all points in the solution domain and the values of $\boldsymbol{F}(\boldsymbol{U}(t-\Delta t), t-\Delta t)$ are known at all interior points. To advance the solution in time to t the fully explicit algorithm proceeds as follows:

Steps 1. Determine an intermediate solution $\boldsymbol{U}_{\boldsymbol{i}}^*(t)$ at all interior nodes using a predictor sub-step

$$\boldsymbol{U}_{\boldsymbol{i}}^*(t) = \boldsymbol{U}_{\boldsymbol{i}}(t-\Delta t) + \alpha\Delta t\boldsymbol{F}_{\boldsymbol{i}}(\boldsymbol{U}_{\boldsymbol{i}}(t-\Delta t), t-\Delta t), \qquad \forall \boldsymbol{i} \in \Omega$$

Steps 2. Determine $\boldsymbol{U}^*(t)$ at the boundary and ghost nodes by solving the boundary conditions

$$\left.\begin{aligned} \boldsymbol{U}_{\boldsymbol{i}}^*(t) - \boldsymbol{u}_B(\boldsymbol{x}_{\boldsymbol{i}}, t) &= 0 \\ \nabla_2\cdot\boldsymbol{U}_{\boldsymbol{i}}^*(t) &= 0 \\ \text{Extrapolate ghost values of } \mathbf{t}_\mu\cdot\boldsymbol{U}_{\boldsymbol{i}}^* & \end{aligned}\right\} \forall \boldsymbol{i} \in \partial\Omega$$

where $\mu = 1, \ldots, n_d - 1$ and only the tangential component of the momentum equation is used.

Steps 3. Determine $P_{\boldsymbol{i}}(t)$ by solving the pressure Poisson equation along with the remaining boundary conditions

$$\nabla_2^2 P_{\boldsymbol{i}}(t) = -\mathcal{J}(\nabla_2 \boldsymbol{U}_{\boldsymbol{i}}^*(t)) + \nabla_2 \cdot \boldsymbol{f}_{\boldsymbol{i}}(t), \qquad \forall \boldsymbol{i} \in \Omega$$

$$\boldsymbol{n} \cdot \nabla_2 P_{\boldsymbol{i}}(t) = -\boldsymbol{n} \cdot \left[\frac{\partial \boldsymbol{U}_{\boldsymbol{i}}^*(t)}{\partial t} + ((\boldsymbol{U}_{\boldsymbol{i}}^*(t) - \boldsymbol{W}_{\boldsymbol{i}}(t)) \cdot \nabla_2)\boldsymbol{U}_{\boldsymbol{i}}^*(t) + \nu \nabla_2 \times \nabla_2 \times \boldsymbol{U}_{\boldsymbol{i}}^*(t) - \boldsymbol{f}(t) \right], \qquad \forall \boldsymbol{i} \in \partial\Omega.$$

The normal component of the momentum equation is used here as a Neumann boundary condition for the pressure Poisson equation. The $\nu\Delta\mathbf{u}$ term has been replaced by $-\nu\nabla \times \nabla\mathbf{u}$ to avoid a viscous time-step restriction, see [55] for more details.

Steps 4. Given $\boldsymbol{U}^*(t)$ and $P(t)$ the pressure gradients can be computed and $\boldsymbol{F}(\boldsymbol{U}^*(t), t)$ found at interior nodes.

Steps 5. Correction steps can now be taken to either increase the time-step, or as needed, to stabilise the algorithm for light rigid bodies. The correction steps consists of the Adams–Moulton corrector for the velocity followed by an additional pressure solve. For light bodies, when added mass effects are large, under-relaxed sub-iterations are used during these corrector steps to stabilise the scheme. Typically 3–7 corrector steps are used in the present work, depending on the significance of added mass effects in the problem.

For moving grids, additional steps in the algorithm are required to evolve the rigid-body equations (as discussed in the next section) and subsequently move the component grids. After the component grids have been moved the overset grid connectivity information is regenerated. Note that since the governing equations are solved in a reference frame moving with the grid, no additional interpolation is needed to transfer the solution at discretisation points from one time step to the next. As grids move, however, some unused points may become active and values at these *exposed-points* are interpolated at previous times as discussed in [31].

For small problems (number of grid points $\mathcal{O}(10^4)$) the linear systems of equations for the velocity components and the separate system of equations for the pressure are effectively solved using direct solution methods. Larger problems necessitate iterative approaches; we use Krylov subspace methods from the PETSc library [56], algebraic multigrid solvers from the Hypre package [57] and the geometric multigrid solvers for overset grids from Overture [58].

6. Fluid–solid coupling

This system of ODEs governing the particle motion is discretised in time using a Leapfrog predictor and Adams–Moulton corrector scheme. The predictor consists of

$$\boldsymbol{v}_b^p = \boldsymbol{v}_b^{n-1} + \frac{2\Delta t}{m_p}\boldsymbol{F}^n, \qquad \boldsymbol{x}_b^p = 2\boldsymbol{x}_b^n - \boldsymbol{x}^{n-1} + \frac{\Delta t}{m_p}\boldsymbol{F}^n,$$

$$\boldsymbol{\omega}^p = \boldsymbol{\omega}^{n-1} + 2\Delta t(-\boldsymbol{\omega}^n \times A\boldsymbol{\omega}^n + \boldsymbol{T}^n), \quad \boldsymbol{e}_i^p = \boldsymbol{e}_i^{n-1} + 2\Delta t(\boldsymbol{\omega}^n) \times \boldsymbol{e}_i^n,$$

and is performed before Step 1 in the time-stepping algorithm of Section 5. The corrector is

$$\boldsymbol{v}_b^{n+1} = \boldsymbol{v}_b^n + \frac{\Delta t}{2m_p}(\boldsymbol{F}^n + \boldsymbol{F}^p), \qquad \boldsymbol{x}_b^{n+1} = \boldsymbol{x}_b^n + \frac{\Delta t}{2m_p}(\boldsymbol{v}_b^n + \boldsymbol{v}_b^p),$$

$$\boldsymbol{\omega}^{n+1} = \boldsymbol{\omega}^n + \frac{\Delta t}{2}(-\boldsymbol{\omega}^n \times A\boldsymbol{\omega}^n + \boldsymbol{T}^n - \boldsymbol{\omega}^p \times A\boldsymbol{\omega}^p + \boldsymbol{T}^p), \quad \boldsymbol{e}_i^{n+1} = \boldsymbol{e}_i^n + \frac{\Delta t}{2}(\boldsymbol{\omega}^n \times \boldsymbol{e}_i^n + \boldsymbol{\omega}^p \times \boldsymbol{e}_i^p)$$

and is performed after Step 3 (pressure solve) in the time-stepping algorithm. A predictor–corrector scheme is used to facilitate the fluid–solid coupling, and to allow for sub-time-step iterations for light bodies as discussed next.

Low solid/fluid density ratios can cause the standard time-stepping routine to become unstable, owing principally to the added-mass instability [40]. To alleviate this, under-relaxed sub-iterations are performed during the correction stages (i.e. fluid velocity solve and pressure solve) of the time-stepping algorithm. These sub-iterations are thus relatively expensive although the implicit systems are not changed during these iterations.

The approach used here is similar to that used by many previous authors, although we prefer to under-relax the force on the rigid-body as opposed to under-relaxing the entire state of the rigid-body. Note that more sophisticated approaches exist to reduce the required number of sub-iterations such as those based on Aitken acceleration [19,59].

We illustrate the relaxed sub-iteration through consideration of the rigid-body velocity equation, though this is performed for the angular velocity equation as well. The force-relaxation sub-iteration replaces the update for $\boldsymbol{v}_b^{n+1}$ in the corrector step above by the iteration

$$\boldsymbol{v}_b^{n+1,k} = \boldsymbol{v}_b^n + \frac{\Delta t}{2}(\boldsymbol{F}^n + \boldsymbol{F}^{n+1,k}), \qquad k = 1, 2, \ldots$$

where k denotes the iteration count. The iterative forcing used to evolve the equation is

$$\boldsymbol{F}^{n+1,k} = (1-\alpha)\boldsymbol{F}^{n+1,k-1} + \alpha\widetilde{\boldsymbol{F}}^k, \qquad \alpha \in (0, 1]$$

where α is a relaxation parameter and $\widetilde{\boldsymbol{F}}^k$ is the forcing at step k, which initially is simply the predicted force from the previous fluid solve step, i.e. $\widetilde{\boldsymbol{F}}^1 = \boldsymbol{F}^p$. During each sub-iteration, the fluid velocity and fluid pressure are recomputed and these updated fluid values are used to compute the next approximation to the force on the rigid body. A small α can ensure stability—at the cost of increased iterations. An optimal value of α is problem dependent and some experimentation is required to reach a good compromise between stability and computational cost. For example, a value of 0.1 was used in the pure wake interaction test case of §7.4 where the maximum number of sub-iterations was 39 during the first few time-steps, likely due to the non-smooth forcing at start up, and the minimum and average number of sub iterations were 5 and 7, respectively. Iterations are performed until the absolute or relative change in the force fall below their respective convergence criteria, $\Delta F^k < \tau_a$, or $\Delta F^k/(|\boldsymbol{F}^k| + \epsilon_F) < \tau_r$, where $\Delta F^k = |\boldsymbol{F}^{n+1,k} - \boldsymbol{F}^k|$.

6.1. Collision model

A hard-sphere collision model based on the linear conservation of momentum is used to handle cases in which particles touch.[5] During the predictor step of the particle advancement scheme the new positions are used to determine whether or not particles breach the minimum separation distance, as stipulated by the requirements of the interpolation stencils. If this minimum separation distance is breached, a collision is deemed to have occurred and the particle velocities are corrected. The velocity corrections are calculated by

$$\hat{\boldsymbol{v}}_{b,A}^{n+1} = \boldsymbol{v}_{b,A}^{n+1} + \left[v_A^n - v_A^{n+1} - \frac{(1+e_r)m_{b,B}}{m_{b,A}+m_{b,B}}(v_A^n + v_B^n) \right] \boldsymbol{n}_A,$$

$$\hat{\boldsymbol{v}}_{b,B}^{n+1} = \boldsymbol{v}_{b,B}^{n+1} + \left[v_B^n - v_B^{n+1} - \frac{(1+e_r)m_{b,A}}{m_{b,A}+m_{b,B}}(v_A^n + v_B^n) \right] \boldsymbol{n}_B,$$

where $v_A = \boldsymbol{v}_{b,A} \cdot \boldsymbol{n}_A$, $v_B = \boldsymbol{v}_{b,B} \cdot \boldsymbol{n}_B$, e_r is the coefficient of restitution and $\boldsymbol{n}_A = -\boldsymbol{n}_B$ is the unit normal vector pointing from the centre of mass of particle A to the centre of mass of particle B. In this work, collisions were modelled as perfectly elastic with a coefficient of restitution of $e_r = 1$. This is a frictionless model, so tangential forces are assumed to be zero during the collision, and angular velocities are not corrected by the model. This hard-sphere model is also restricted to the contact of only two particles at any given moment in time.

7. Numerical results

7.1. Convergence study

To accurately simulate viscous flows the grid resolution must be fine enough to fully capture boundary layers adjacent to solid surfaces. These can be very thin, depending on the Reynolds number of the problem as the boundary layer depth scales approximately as $1/\sqrt{\mathrm{Re}}$, see [60]. A major advantage of boundary-conformal over static grid methods is the ability to selectively refine the grid near solid boundaries. In a detailed grid independence study of viscous flow past a static cylinder, Nicolle [61] investigated how refinement of different areas of the grid affected the behaviour of the cylinder. Predictably, the surface resolution was found to most affect the cylinder behaviour, but the downstream wake resolution was found to affect the Strouhal number. Nonetheless, large ratios between surface and wake resolution were found to give very accurate results.

In the present work, emphasis is placed on the grid characteristic length scales to optimise run time. Following Nicolle and Eames [62] we use two grid length scales to quantify the quality of the computational grid: the domain length scale (*DLS*) is the background grid element size while the surface length scale (*SLS*) is the grid element size on the surface of the particle.

The grid independence study is first performed using a grid with nearly uniform grid spacings and is then repeated using grid refinement near the particle boundary. Descriptions of the grids used are provided in Table 1. Because curvilinear grids are used to represent the particles, the cells are slightly distorted in physical space. Thus, Table 1 provides minimum, average and maximum cell volumes (areas).

The test used in both convergence studies is as follows: the domain is a rectangular channel of dimensions $(W^*, H^*) = (4\,D, 40\,D)$ filled with an a priori quiescent fluid of density $\rho_f = 1$ g/cm^3 and kinematic viscosity of 0.05 cm^2/s. The particle $(D, \rho_p) = (0.25, 1.5)$ is released from rest at $(x_0^*, y_0^*) = (D, 38.4\,D)$ with the gravitational constant set to $\boldsymbol{g} = 981$ cm/s^2 in the negative y-direction. The problem geometry can be seen in Fig. 2. The results are non-dimensionalised as follows:

$$u^* = \frac{u}{U_T}, \quad v^* = \frac{v}{U_T}, \quad x^* = \frac{x}{D}, \quad y^* = \frac{y}{D}, \quad \omega^* = \frac{\omega D}{U_T}, \quad t^* = \frac{tU_T}{D} \tag{9}$$

where $U_T = 8.6041$ cm s^{-1} is the measured terminal settling velocity.

Table 1
Description of the uniform and refined grids used in the convergence study.

Grid	Node count	Cell volume			*DLS*	*SLS*
		Min.	Ave.	Max.		
GU1	1497931	0.66	0.678	0.678	$D/96$	$D/96$
GU2	670984	1.47	1.53	1.53	$D/64$	$D/64$
GU3	380394	2.58	2.71	2.71	$D/48$	$D/48$
GU4	171724	5.65	6.10	6.10	$D/32$	$D/32$
GU5	98102	9.90	10.8	10.9	$D/24$	$D/24$
GU6	44967	21.3	24.4	24.4	$D/16$	$D/16$
GR1	679685	0.67	1.52	1.53	$D/64$	$D/96$
GR2	390285	0.78	2.68	2.71	$D/48$	$D/96$
GR3	191265	0.78	5.68	6.10	$D/32$	$D/96$
GR4	123095	0.71	9.09	10.9	$D/24$	$D/96$
GR5	63935	0.78	18.2	24.4	$D/16$	$D/96$
GR6	31455	0.78	39.5	97.7	$D/8$	$D/96$

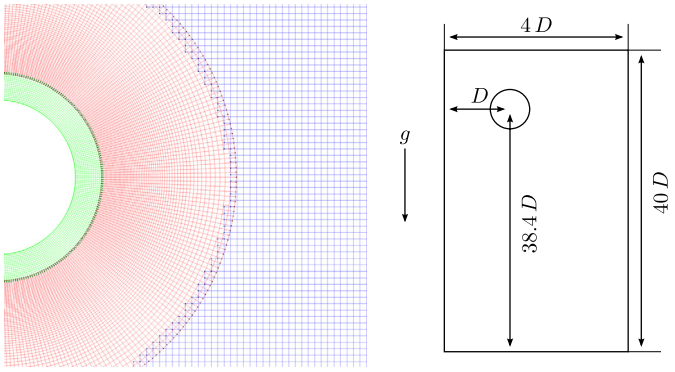

Fig. 2. Left: Cropped view of *GR4*, showing the boundary layer grid (green), transition grid (red) and background grid (blue) with interpolation points. Right: Geometry of the convergence study problem. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

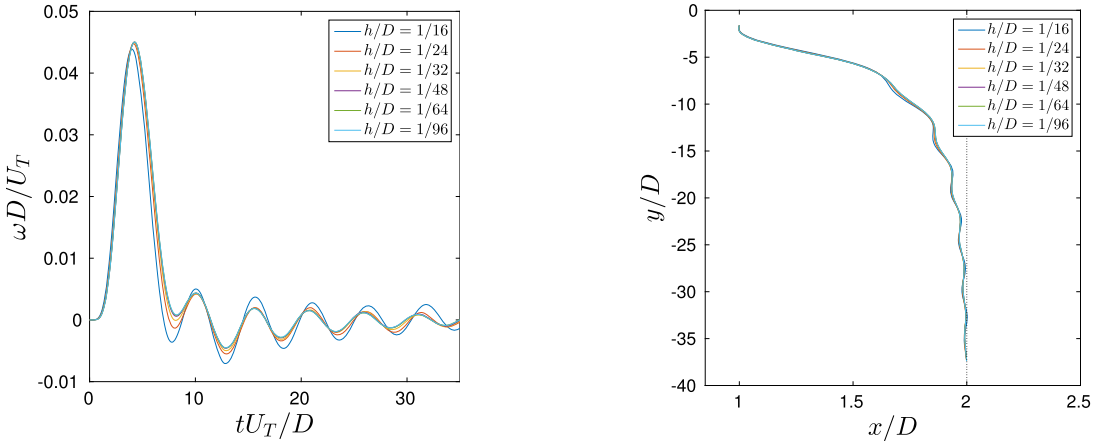

Fig. 3. Left: Normalised angular velocity history for the convergence study at increasing grid resolutions. Right: Position of the disk in the channel for the convergence study at increasing grid resolutions.

Under the action of gravity the particle rotates in a clockwise sense, as if rolling down the wall. Immediately after release it moves a short distance towards the near wall before migrating towards its equilibrium position along the channel centreline.

The wake remains attached but is unsteady. This is reflected in the oscillatory u^* and ω^* velocity time histories, shown in Figs. 4 and 3. The angular velocity time history exhibits a large initial peak after which it is rapidly damped to low

[5] Note that in principle the particles should never actually touch, but resolving the near contact would require a very fine grid.

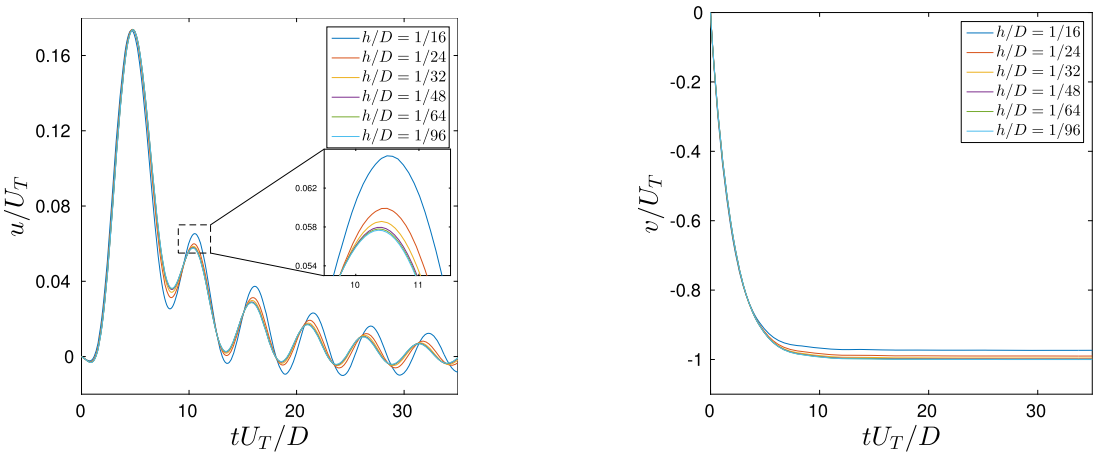

Fig. 4. Left: Normalised horizontal velocity history for the convergence study at increasing grid resolutions. Right: Normalised vertical velocity history for the convergence study at increasing grid resolutions.

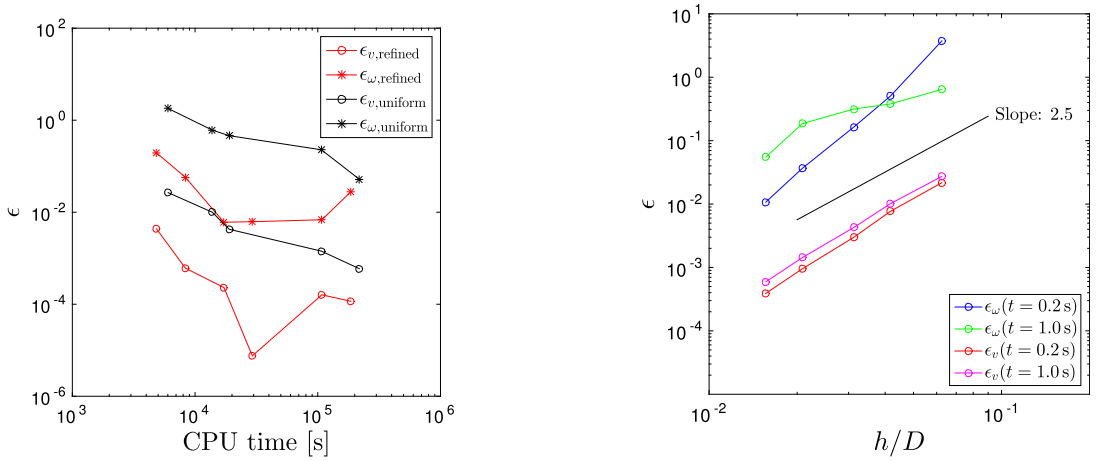

Fig. 5. Left: Comparison of relative error magnitude in v^* and ω^* at $t = 1$ s against required CPU time using uniform and refinement grids. Right: relative vertical and angular velocity errors at early ($t = 0.2$ s) and late ($t = 1.0$ s) stages of the simulation.

amplitude oscillations about $\omega^* = 0$. After a very small negative peak, the u^* time history exhibits a large positive peak, much like ω^*, with damped successive peaks. However, unlike in ω^*, the following u^* oscillations have a non-zero mean value as the disk drifts towards its equilibrium position. In contrast, the vertical velocity, v^*, shows the particle rapidly reaching a steady settling velocity, unaffected by the attached unsteady wake.

Fig. 5 show relative *errors* (meaning differences compared to the reference solution) in v^* and ω^*, which were calculated as $\epsilon_v = (v^* - v^*_{\text{ref}})/v^*_{\text{ref}}$ and $\epsilon_\omega = (\omega - \omega^*_{\text{ref}})/\omega^*_{\text{ref}}$, where data from the *GU1* simulation are used as reference values. Relative errors in v^* and ω^* taken early on in the simulation, at $t = 0.2$ s, and late in the simulation, at $t = 1.0$ s, show greater than second order convergence for both components early on but a decrease in convergence rate for ω^* as the simulation progresses. The gravitational acceleration is impulsively turned on at $t = 0$, and this non-smooth forcing could have a detrimental effect on the convergence rates.

The test was repeated using a series of refined grids. These were constructed using a fine grid immediately surrounding the particle surface, smoothly connected to a coarse background grid using a transition grid. This construction can be seen in Fig. 2. Detailed descriptions of the grids are provided in Table 1, where uniform grids are denoted by the prefix *GU* and refinement grids by *GR*.

Table 2 shows absolute values of u^*, v^* and ω^* at $t = 1.0$ s as well as relative errors, where data from *GU1* were taken as reference values. As before, it is evident that v^* is fairly insensitive to the grid resolution, while u^* and ω^* show a large dependence on the near surface resolution. With a high resolution surface grid capturing the boundary layer, the motion of the disk can be captured quite accurately, even with a large surface to background grid resolution ratio. In fact, the *GR5* grid with a resolution ratio of 6 : 1 reproduced solutions of *GU1* with a maximum error of 5%, with a more than 23 fold reduction in number of grid points. Fig. 5 shows the relative error in v^* and ω^* at $t = 1$ s against the total CPU time of the calculation.

Table 2
Absolute values and relative errors for the convergence study taken at $t = 1.0$ s.

Grid	v/U_T	u/U_T	$\omega D/U_T \times 10^{-4}$	$\epsilon_v \times 10^{-4}$	ϵ_u	ϵ_ω
GU1	1.0000	0.0033	6.4761	–	–	–
GU2	0.9994	0.0034	6.1663	5.9299	0.0288	0.0521
GU3	0.9994	0.0034	6.8561	14.000	0.0969	0.2275
GU4	0.9957	0.0041	9.2127	43.000	0.2775	0.4609
GU5	0.9900	0.0046	9.4015	100.00	0.4513	0.6093
GU6	0.9734	0.0052	0.0016	266.00	0.6329	1.7913
GR1	0.9995	0.0034	6.5961	1.1599	0.0292	0.0277
GR2	0.9998	0.0033	6.4250	1.6028	0.0141	0.0069
GR3	1.0000	0.0031	6.4250	0.0755	0.0186	0.0062
GR4	1.0002	0.0031	5.8773	2.3025	0.0161	0.0061
GR5	1.0006	0.0032	6.1766	6.0949	0.0013	0.0573
GR6	1.0043	0.0026	6.9848	43.000	0.1915	0.1956

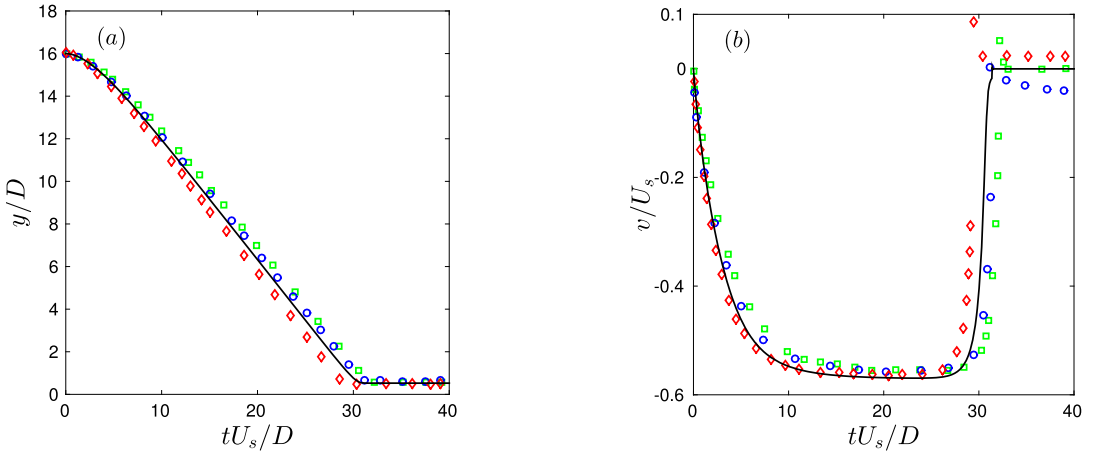

Fig. 6. Histories of the y^*-coordinate and v^* component of the centre of the disk for a low Reynolds number sedimentation of a symmetrically placed disk test case with data from Hu et al. [64] (□), Wang et al. [63] (○), Glowinski et al. [13] (◇) and the present study (−).

7.2. Settling disk impacting a wall

This test simulates the fall of a rigid circular disk in a bounded domain and its impact with the bottom boundary. This test has been performed by other researchers using DLM/FDM [13], an FEM fictitious boundary method [63], and an immersed boundary lattice Boltzmann method [64]. The computational domain has a width of $W = 8\,D$, a height of $H = 24\,D$ and grid characteristic length scales $DLS = D/16$ and $SLS = D/96$, where $D = 0.25$ cm is the disk diameter. The disk is initially placed along the centreline of the domain, $8\,D$ from the top boundary. The disk has density $\rho_d = 1.25$ g/cm^3 and the kinematic viscosity of the fluid is $\nu = 0.1$ cm^2/s. The results are non-dimensionalised as in (9), where the characteristic velocity scale U_s is an estimated terminal velocity,

$$U_s = \sqrt{\frac{\pi D}{2}\left(\frac{\rho_d - \rho_f}{\rho_f}\right) g}. \tag{10}$$

The present results (Fig. 6) are in good agreement with the previous studies. The disk reaches the terminal settling velocity at $t^* = 20$, with a terminal particle Reynolds number of $\mathrm{Re}_T = 17.45$, consistent with the literature. As the disk approaches the bottom wall the results differ slightly. The studies compared against in Fig. 6 all exhibit a rebound of the disk from the bottom boundary while the present results do not.

In the present study the grid around the disk and along the bottom of the tank is very fine ($SLS = D/96$ for both the disk and the bottom of the tank), allowing the lubrication forces and flow in the gap to be better resolved. This slows the particle down more before "contact" is made with the wall.[6] Additionally, the present study used a conservation of linear momentum based hard-sphere collision model approach to model the collision between the disk and the bottom boundary. The previous studies compared against here all used repulsive potential type methods. We can estimate a Stokes number for the particle to comment on the "correctness" of the present results. The Stokes number, Stk, is the ratio between the

[6] In fact an infinitely smooth disk with flow governed by the Navier–Stokes equations should never actually contact the wall but in this settling case only approach the wall exponentially slowly with the gap becoming ever thinner.

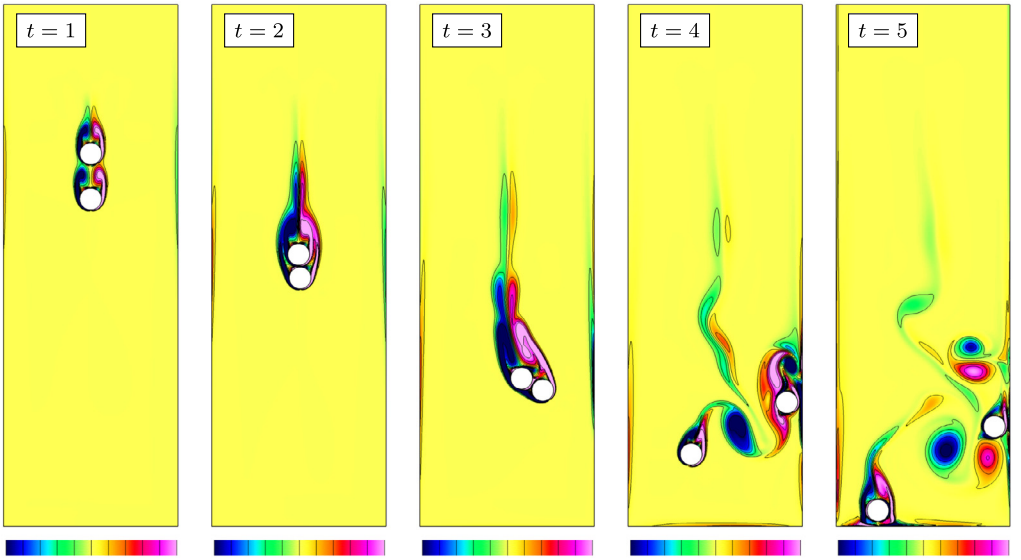

Fig. 7. Settling of two disks in a quiescent fluid. Contours of the vorticity at five different time, with a vorticity scale between $-3.6 \leq \xi D/U_s \leq 3.6$.

particle and fluid relaxation times, τ_p and τ_f, respectively. Taking $\tau_f = R/U_T$, where R is the disk radius, $\tau_p = mv/F_d$, and F_d is the drag force on the disk, then $\text{Stk} = mv^2/(RF_d)$. Once the disk reaches its terminal settling velocity the drag force balances with the force due to gravity, so $F_d = \pi R^2 g(\rho_d - \rho_f)$ and thus the Stokes number is $\text{Stk} = v^2/(Rg(1 - \frac{\rho_f}{\rho_d}))$. Here, the Stokes number is approximately 2. It has been demonstrated that for 3D cases particles settling with $\text{Stk} < 10$ there is no rebound after contact is made with the bottom boundary [65,66]. Assuming this holds true for the 2D equivalent, then the above results indicate that the repulsive potential collision model is a poor sub-grid model for low speed impacts. The rebound evident in the study of Glowinski et al. [13] indicates that a higher grid resolution is required to adequately resolve the lubrication forces than the hydrodynamic interactions during free-fall. While the current approach is adequate here it is clear that in many situations resolving the gap is not practical and prohibitively expensive. Qiu et al. [67] presented a novel solution to computing incompressible flow in thin gaps using pressure degrees of freedom on virtual solid surfaces to provide solid–fluid coupling in the gap region, which when extended to no-slip boundaries could be a good alternative for this sort of problem.

7.3. Settling of two offset disks

Two offset cylinders settling in a quiescent fluid are simulated to demonstrate the drafting, kissing, tumbling behaviour observed experimentally by Fortes et al. [68]. This is a difficult problem to simulate owing to the non-linear nature of the particle motion and the particle–particle and particle–wall interactions. Results are compared to previous studies of Patankar et al. [69,70], Wan and Turek [63], Niu et al. [71], Zhang and Prosperetti [72] and Feng and Michaelides [73] for a low Reynolds number case and Uhlmann [16] for a moderate Reynolds number case. The low Reynolds number case uses a computational domain of width $10\,D$ and height $40\,D$, with the particles of diameter 0.2 cm placed along the vertical centreline, $4\,D$ and $6\,D$ from the top boundary. The high Reynolds number case uses a computational domain of width $8\,D$ and height $24\,D$, with the particles of diameter 0.25 cm placed $4\,D$ and $6\,D$ from the top boundary, and offset by $D/250$ and $-D/250$ from the vertical centreline. For the low Reynolds number case, the grid characteristic length scales are $DLS = D/19$ and $SLS = D/76$ whilst for the moderate Reynolds number case $DLS = D/24$ and $SLS = D/128$. In each case, both the top and bottom particles have the same density ratio. For the low Reynolds number case, the density ratio is $\rho_r = 1.01$ and the fluid has kinematic viscosity $\nu = 0.1$ cm^2/s. For the moderate Reynolds number case, the density ratio is $\rho_r = 1.5$ and the kinematic viscosity is $\nu = 0.01$ cm^2/s. In both cases the gravitational constant was taken as $\boldsymbol{g} = 981$ cm/s^2 and the results are non-dimensionalised as in (9), where the characteristic velocity, U_s, is again calculated using (10).

Fig. 7 shows the positions of the particles as they sediment, interacting with each other and the domain boundary, along with instantaneous vorticity contours. The observed dynamical interactions are in good agreement with those observed in the quasi two-dimensional experiments in [68]. Initially, the two particles begin moving from rest under the influence of gravity with the same acceleration. As the wake forms behind the lower particle the top particle becomes shielded in the resultant low pressure region. This allows the top particle to draft behind the lower particle, similar to cyclists in a peloton. This is the “drafting” stage. Eventually, the top particle makes near contact with the lower particle (they “kiss”) and effectively form an elongated body with axis parallel to the fall. This configuration is inherently unstable and the elongated body rotates to align its long axis perpendicular to the fall. This is the “tumbling” stage described in [68]. The particles

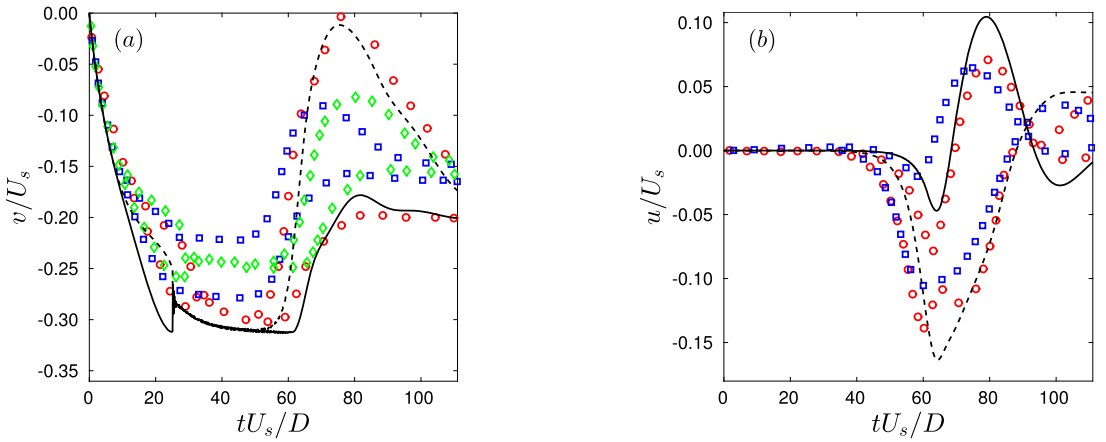

Fig. 8. Histories of the (*a*) v^* and (*b*) u^* velocity components of the centre of the disks for the low Reynolds number drafting, kissing, and tumbling test case, with $\rho_d = 1.01$, $\nu = 0.1$ where the solid line denotes the (initially) top disk and the dashed line the bottom disk, with data from: (*a*) Patankar [70] (◇), Patankar et al. [69] (□) and Feng et al. [73] (○); (*b*) Patankar et al. [69] (□) and Feng et al. [73] (○) overlayed.

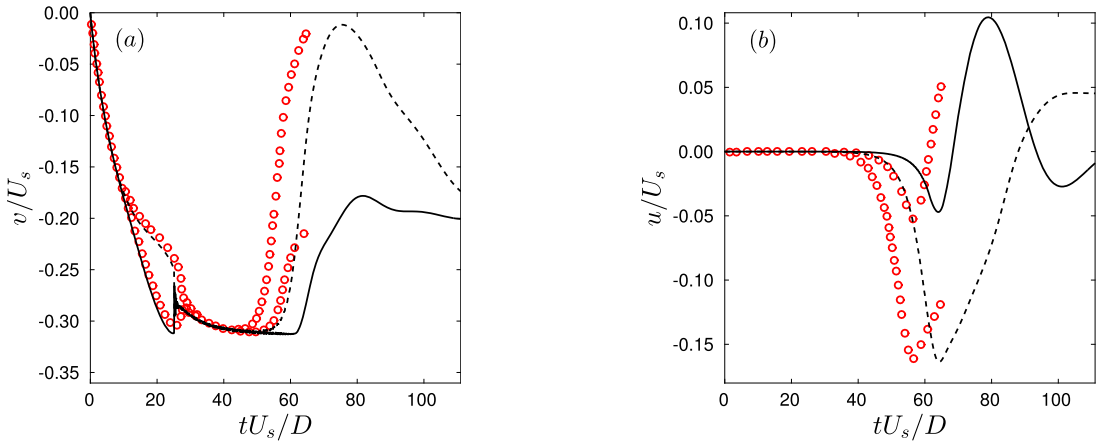

Fig. 9. Time histories of the (*a*) v^* and (*b*) u^* velocity components of the centre of the disks for the low Reynolds number drafting, kissing, and tumbling test case with $\rho_d = 1.01$, $\nu = 0.1$, where the solid line denotes the (initially) top disk and the dashed line the bottom disk, and data from Zhang et al. [72] overlayed (○).

separate and the lower particle is overtaken by the top particle, which continues to sediment with a slightly negative u^* velocity. The other particle impacts the wall, after which it, too, sediments with a slightly negative u^* velocity.

The results for the low Reynolds number case compare well qualitatively with those of [69,70,73] but not quantitatively (see Fig. 8). Though quantitative agreement is not necessarily apparent amongst the results of these studies themselves, what is apparent is that their settling velocities are all lower than those found in the present study. Although different methods were used, all three of these simulations used low grid resolutions around the particles, particularly [73]. The study in [72] used a higher resolution grid, with 20 computational nodes per particle diameter. Very good quantitative agreement is found between that study and the present one, as is evident from Fig. 9, though there is a discrepancy in the duration of the "kissing" contact and the onset of "tumbling". The onset of "tumbling" is caused by the build up of numerical error, so this is expected to be solver specific. In the absence of numerical error, or bias introduced by the grid, the disks would not leave the "kissing" stage [16].

Results for the moderate Reynolds number case are shown in Fig. 10. These are compared to results from [16], who used an immersed boundary method on high resolution grids. Both qualitatively and quantitatively the results are in excellent agreement for the particle positions and u, v velocity components, with the only differences found during the initial contact and subsequent "kissing" stage, due to the different collision models. All of the aforementioned studies used a repulsive force based model, while a conservation of linear momentum model is used here. Fig. 10 shows good qualitative but poor quantitative agreement for the angular velocity component. This is a very sensitive metric [16] and it is likely that the differences are due, in large part, to the different collision mechanisms used. The novel approach of Kempe and Fröhlich [21], which uses a sub-grid lubrication force correction and conserves angular momentum, would probably be a more appropriate collision mechanism for this case.

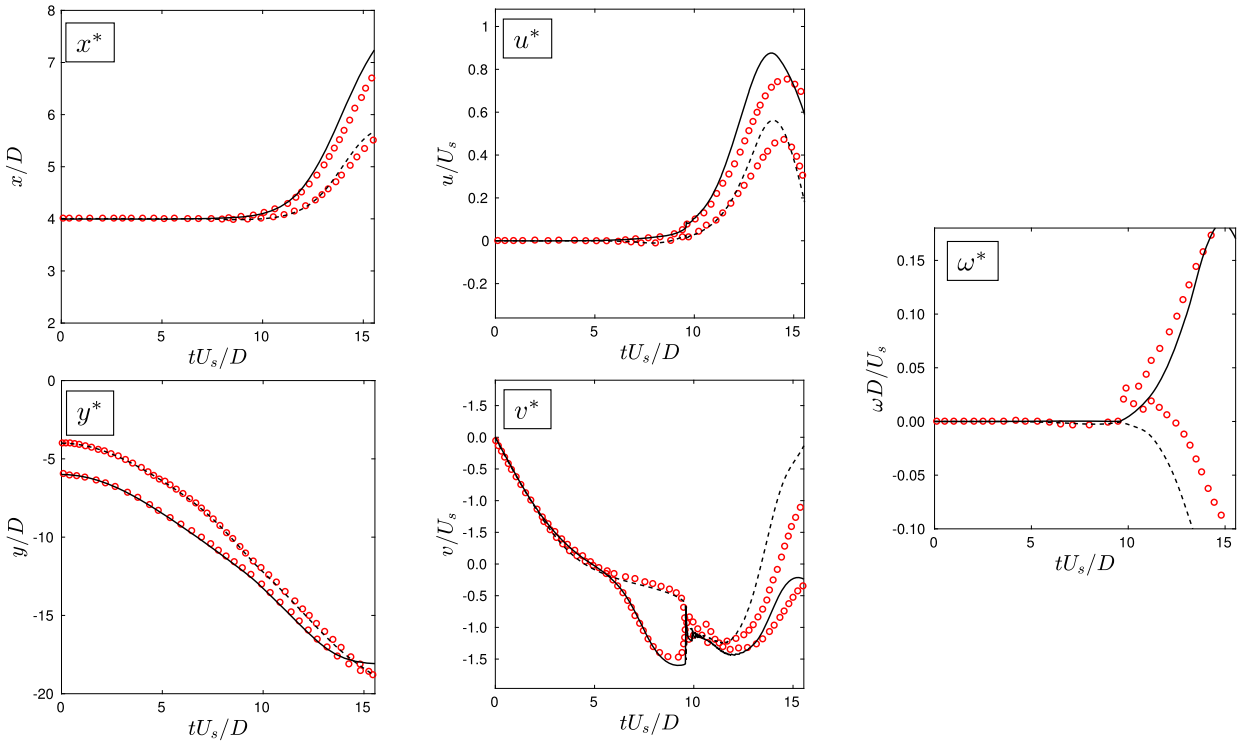

Fig. 10. Time histories of x^*, y^*, v^*, u^* and ω^* for the moderate Reynolds number drafting, kissing, and tumbling test case with $\rho_d = 1.5$, $\nu = 0.01$, where the solid line denotes the (initially) top disk and the dashed line the bottom, and data from Uhlmann [16] overlayed (○).

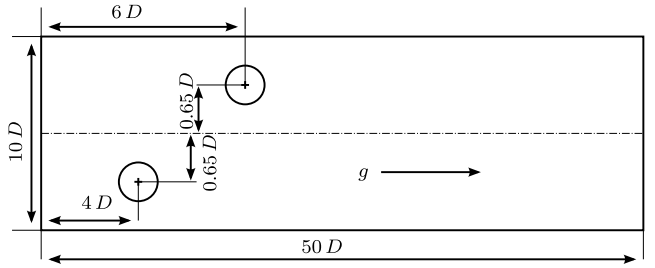

Fig. 11. Problem geometry for the two particle wake interaction test case.

7.4. Two particle wake interaction

This is a case presented by Uhlmann [16] to test the fluid–structure interaction, with particular emphasis on examining the effect of wake interactions between the particles on the angular velocity. Two particles of differing densities settle in an otherwise quiescent fluid. The heavier particle passes the lighter particle, subjecting it to perturbations from its wake. The particles do not collide and therefore no collision model is required, making it an attractive benchmark case.

The computational domain, shown in Fig. 11, has a width of $10\,D$, a height of $50\,D$ and the grid characteristic length scales are $DLS = D/40$ and $SLS = D/100$, where the particle diameter is $D = 0.2$ m. A heavier particle of density ratio $\rho_{r,1} = 1.5$ is initially positioned at $\boldsymbol{x} = (-0.65\,D, 4\,D)$ from the channel centreline and top boundary respectively, while the lighter particle of density ratio $\rho_{r,2} = 1.25$ is positioned at $\boldsymbol{x} = (0.65\,D, 6\,D)$. Both particles are initially at rest and the fluid of kinematic viscosity $\nu = 0.0008$ m^2/s is quiescent at $t = 0$ s. The gravitational constant is set to $\boldsymbol{g} = 9.81$ m/s^2. The results are non-dimensionalised as in (9), where the characteristic velocity U_s is calculated using (10) and the density ratio of the heavier disk, viz. $\rho_{r,1} = 1.5$.

The maximum particle Reynolds numbers of the heavy and light particle are 280 and 230, respectively [16]. Uhlmann [16] used a uniform grid resolution of $DLS = D/40$, allowing for fewer than three grid points across the estimated boundary layer depth for both the light and heavy particles. Given the findings of the convergence study in section 7.1 it is not likely that the solutions at this grid resolution are grid independent. A brief convergence study was performed, shown

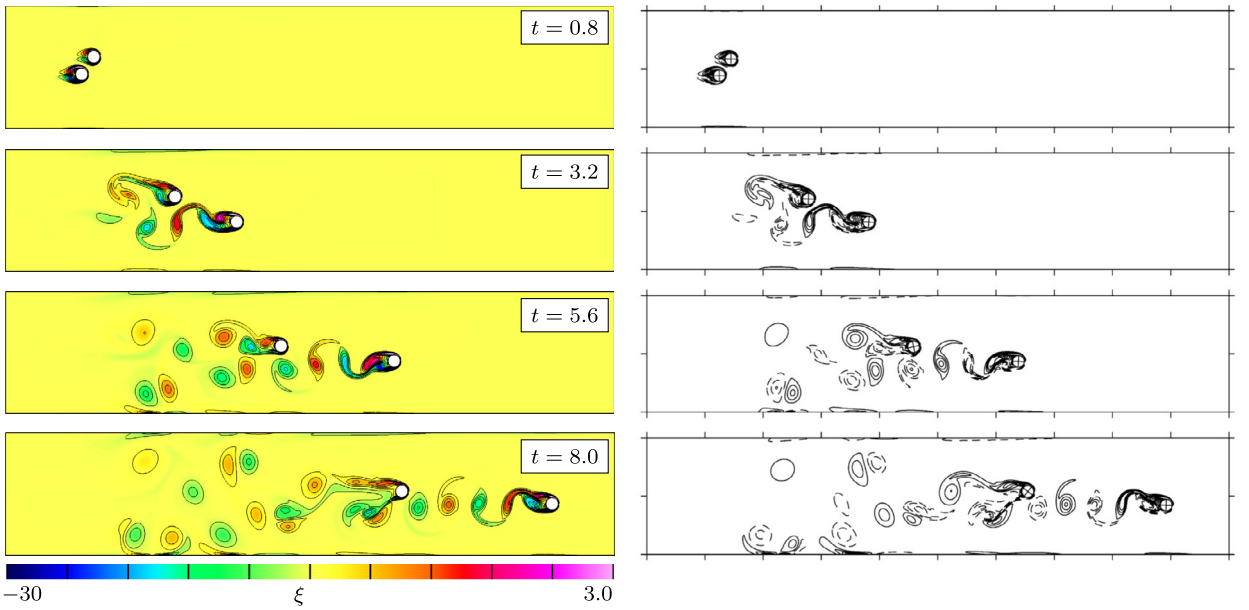

Fig. 12. Contours of vorticity at times $t = 0.8$, $t = 3.2$, $t = 5.6$, $t = 8.0$ for the pure wake interaction test case compared to plots from Uhlmann [16] taken at the same times and with the same vorticity extrema. Left: present study using a quasi-uniform grid with $DLS = D/40$ and $SLS = D/40$. Right: results from [16] computed on a uniform grid of resolution $D/40$.

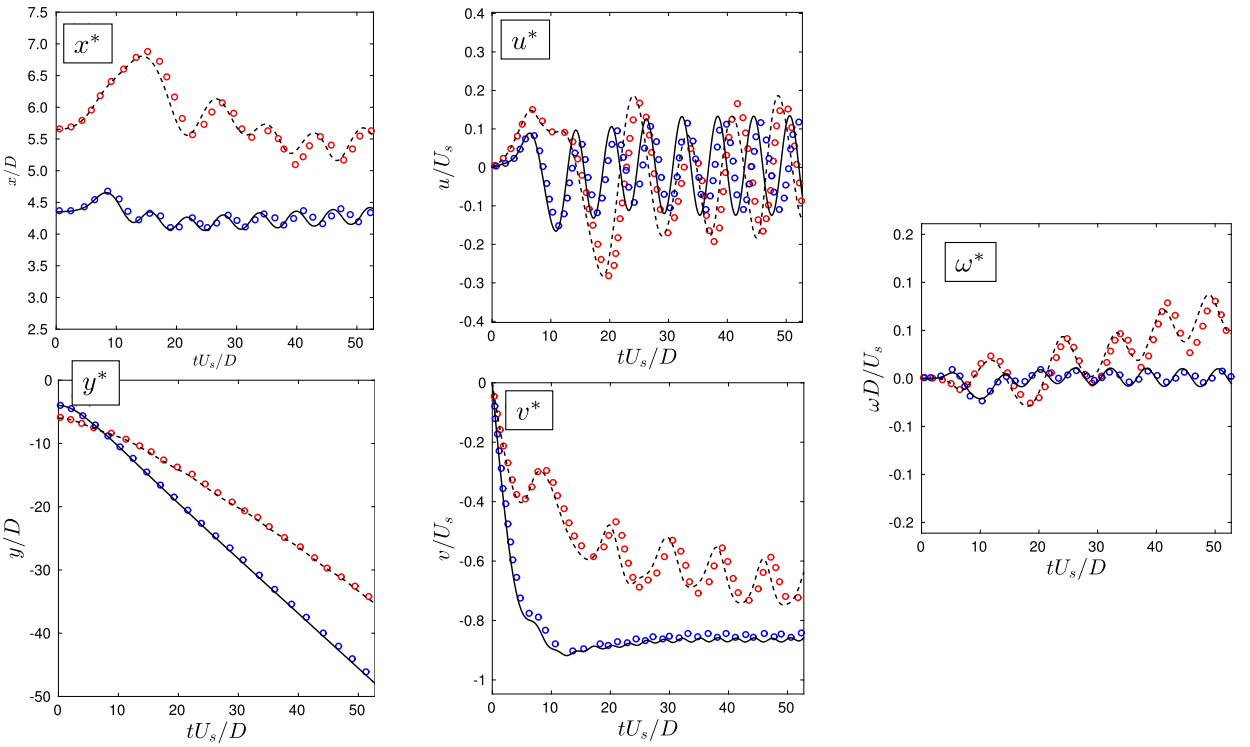

Fig. 13. Time histories of x^*, y^*, v^*, u^* and ω^* for the pure wake interaction test case with $\rho_{r,1} = 1.5$, $\rho_{r,2} = 1.25$, $\nu = 0.0008$ m^2/s and data from Uhlman [16] overlayed.

in Table 3, and solutions were found to converge at $SLS = D/100$, allowing for 6 points across the boundary layers for both the light and heavy particle.

Fig. 12 shows successive snapshots of the instantaneous vorticity field with snapshots from [16] below. The evolving flow field and particle positions match well. Fig. 13 shows the time histories of the particles position, velocity and angular

Table 3
Absolute values and relative errors for disk 2 taken at $t = 1$ s of the wake interaction grid independence study.

SLS	u/U_s	v/U_s	$\omega D/U_s$	ϵ_u	ϵ_v	ϵ_ω
D/200	0.14550	−0.36470	−0.01123	–	–	–
D/150	0.14554	−0.36483	−0.01225	0.00028	0.00033	0.00181
D/100	0.14567	−0.36518	−0.01132	0.00121	0.00130	0.00831
D/50	0.14680	−0.36705	−0.01205	0.00896	0.00643	0.07353
D/40	0.01479	−0.36847	−0.01293	0.01644	0.01033	0.15205

Table 4
Parameters used for the four settling sphere cases.

Case number	ρ_f [kg/m^3]	μ_f [N s/m^2]	*Re*	Stk
1	970	0.373	1.5	0.19
2	965	0.212	4.1	0.53
3	962	0.113	11.6	1.50
4	960	0.058	31.9	4.13

velocity. The converged solutions of the present study match well with those in [16], although a phase shift is apparent in the oscillatory components and the settling velocity is slightly higher in the present study.

The largest differences are found in the horizontal velocity components, particularly for the light particle. These differences are probably due in large part to the differences in the angular velocity components, which will affect vortex shedding and lift on the particles. The amplitudes of the angular velocity component oscillations for the heavier particle match well but a slight phase shift is apparent. This is reflected in the horizontal velocity components for the heavier particle by matching amplitudes but markedly different periods of oscillation. The angular velocity components of the lighter particle differ in both amplitude and period of oscillation, leading to more pronounced differences in the horizontal velocity components of the two studies.

7.5. Settling sphere

As a final validation case we compare experimental [74] and numerical [24] results on the motion of a single sphere in a closely confined container to numerical results produced by the current method. The sphere of diameter $D = 15$ mm and density $\rho = 1120$ kg/m^3 is positioned centrally with the bottom of the sphere 120 mm from the bottom of the tank, which has *depth* × *width* × *height* dimensions of $100 \times 100 \times 160$ mm. Four cases were run, with Reynolds numbers ranging from 1.5 to a moderate 31.9. The material parameters used in each case are detailed in Table 4, with $\mathbf{g} = 9.81$ m/s^2 throughout. For each case the grid characteristic length scales are $DLS = D/10$ and $SLS = D/38$, allowing for approximately six points across the estimated boundary layer depth for the highest Reynolds number case.

The sphere undergoes three distinct periods of motion after its release from rest: an initial acceleration followed by a period of steady fall at a terminal settling velocity and finally a deceleration as it approaches the wall. As the Reynolds number is increased the three stages become progressively shorter. Similar to Yang and Stern [24] the wall collisions were not considered here and the simulations stopped before the sphere made contact with the wall.

The present results are shown in Fig. 14 and match satisfactorily with the experimental results of ten Cate et al. [74] although some slight differences remain: the terminal settling velocity in case 1 is found to be approximately 4.6% lower here and in case 2 the sphere begins the wall induced deceleration sooner than in [74]. This earlier deceleration in case 2 is also present in the results of [24], as is the lower terminal settling velocity of case 1. The benefit of using overset grids is again evident; the grid used above consists of 5.18×10^5 grid points while the results are as good as those produced on a uniform grid over three times the size.

8. Conclusion

We evaluated the overset grid method for DNS of viscous, incompressible fluid flow with rigid, moving bodies. Several FSI benchmark test cases were carried out for verification and validation purposes. A systematic convergence test was carried out using six uniformly refined grids and six with local refinement near the particle surface. Local refinement was found to produce results deviating no more than five percent from the reference solutions, with a more than 23 fold decrease in grid point count and a subsequent 13 fold decrease in CPU time.

Results compared favourably with those from the literature for the symmetrically placed disk settling in a tank. Discrepancies in the approach and rebound behaviour are due to the hard-sphere collision model and the selective grid refinement, allowing lubrication forces to be better resolved in the current study. The present method compared well with high resolution studies for the dropping, kissing, tumbling test cases and the pure wake interaction test case. Finally, results for a sphere settling in a small tank at various Reynolds numbers compared well with both experimental results of ten Cate et al. [74] and recent numerical results of Yang et al. [24], using only one third the number of grid points as the latter study.

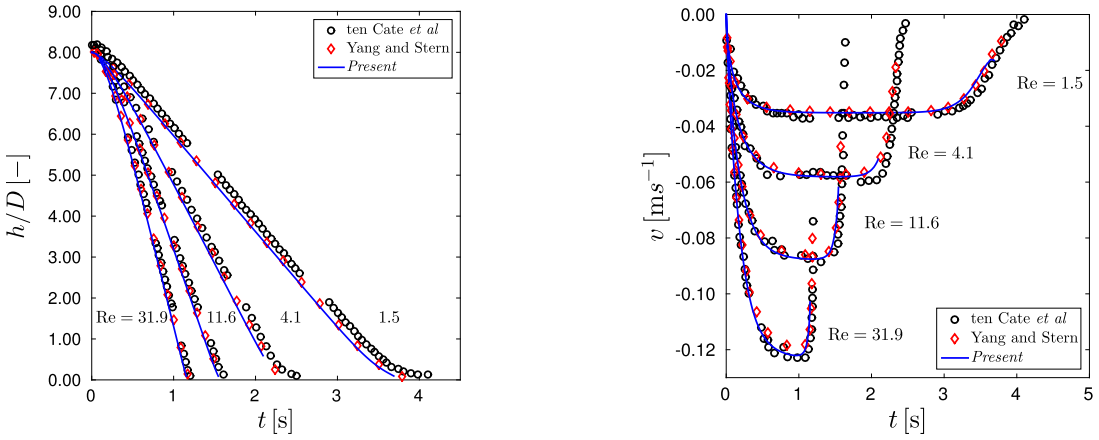

Fig. 14. Left: non-dimensionalised vertical position of the sphere compared to numerical results of Yang et al. [24] (◊) and experimental results of ten Cate et al. [74] (○). Right: dimensional vertical velocity of the sphere compared to numerical results of [24] (◊) and experimental results of [74] (○).

The popular test cases presented in this work are all, to varying degrees, inertially dominated and exhibit viscous boundary layers that must be fully resolved to accurately simulate the behaviour of the rigid bodies in the flow. The second-order accurate boundary fitted method demonstrated here was found to produce reasonably converged results with approximately six grid points across the estimated boundary layer depth. By using a coarse—but fine enough to resolve wake structures—Cartesian background grid and refined, boundary-fitted grids, grid point counts were greatly reduced, even in two-dimensional problems.

The overset grid method has shown promising capabilities for fully-resolved DNS of small numbers of rigid particles. With a more sophisticated collision mechanism, e.g. the multi-scale approach of Kempe et al. [75] or the DEM approach of Wachs [76], fully-resolved DNS of larger numbers of arbitrarily shaped particles could be performed. Without modification to the underlying discretisation technique other types of flow, for example arbitrarily moving bodies in non-Newtonian flows, could be examined.

References

[1] E.B. Nelson, D. Guillot, Well Cementing, 2nd ed., Schlumberger, 2006.
[2] P. Bagchi, Mesoscale simulation of blood flow in small vessels, Biophys. J. 92 (2007) 1858–1877.
[3] A.S. Sangani, A.K. Didwania, Dynamic simulations of flows of bubbly liquids at large Reynolds numbers, J. Fluid Mech. 250 (1993) 307–337.
[4] V.I. Kushch, A.S. Sangani, P.D.M. Spelt, D.L. Koch, Finite-Weber-number motion of bubbles through a nearly inviscid liquid, J. Fluid Mech. 460 (2002) 241–280.
[5] J.F. Brady, Stokesian dynamics, Annu. Rev. Fluid Mech. 20 (1988) 111–157.
[6] N. Takashi, T.J. Hughes, An arbitrary Lagrangian–Eulerian finite element method for interaction of fluid and a rigid body, Comput. Methods Appl. Mech. Eng. 95 (1) (1992) 115–138.
[7] H.H. Hu, N.A. Patankar, M.Y. Zhu, Direct numerical simulations of fluid–solid systems using the arbitrary Lagrangian–Eulerian technique, J. Comput. Phys. 169 (2001) 427–462.
[8] J. Vierendeels, K. Dumont, E. Dick, P. Verdonck, Analysis and stabilization of fluid–structure interaction algorithm for rigid-body motion, AIAA J. 43 (12) (2005) 2549–2557.
[9] M. Coquerelle, G.-H. Cottet, A vortex level set method for the two-way coupling of an incompressible fluid with colliding rigid bodies, J. Comput. Phys. 227 (21) (2008) 9121–9137.
[10] F. Gibou, C. Min, Efficient symmetric positive definite second-order accurate monolithic solver for fluid/solid interactions, J. Comput. Phys. 231 (8) (2012) 3246–3263.
[11] R. Glowinski, T.-W. Pan, T.I. Hesla, D.D. Joseph, A distributed Lagrange multiplier/fictitious domain method for particulate flows, Int. J. Multiph. Flow 25 (5) (1999) 755–794.
[12] R. Glowinski, T.-W. Pan, T.I. Hesla, D.D. Joseph, J. Périaux, A distributed Lagrange multiplier/fictitious domain method for the simulation of flow around moving rigid bodies: application to particulate flow, Comput. Methods Appl. Mech. Eng. 184 (2) (2000) 241–267.
[13] R. Glowinski, T.W. Pan, T.I. Hesla, D.D. Joseph, J. Périaux, A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: application to particulate flow, J. Comput. Phys. 169 (2001) 363–426.
[14] S.D. Costarelli, L. Garelli, M.A. Cruchaga, M.A. Storti, R. Ausensi, S.R. Idelsohn, An embedded strategy for the analysis of fluid structure interaction problems, Comput. Methods Appl. Mech. Eng. 300 (2016) 106–128.
[15] T. Kajishima, S. Takiguchi, Interaction between particle clusters and particle-induced turbulence, Int. J. Heat Fluid Flow 23 (5) (2002) 639–646.
[16] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, J. Comput. Phys. 209 (2005) 448–476.
[17] D. Kim, H. Choi, Immersed boundary method for flow around an arbitrarily moving body, J. Comput. Phys. 212 (2006) 662–680.
[18] T.-R. Lee, Y.-S. Chang, J.-B. Choi, D.W. Kim, W.K. Liu, Y.-J. Kim, Immersed finite element method for rigid body motions in the incompressible Navier–Stokes flow, Comput. Methods Appl. Mech. Eng. 197 (25) (2008) 2305–2316.
[19] I. Borazjani, L. Ge, F. Sotiropoulos, Curvilinear immersed boundary method for simulating fluid structure interaction with complex 3D rigid bodies, J. Comput. Phys. 227 (16) (2008) 7587–7620.
[20] W.-P. Breugem, A second-order accurate immersed boundary method for fully resolved simulations of particle-laden flows, J. Comput. Phys. 231 (13) (2012) 4469–4498.

[21] T. Kempe, J. Fröhlich, An improved immersed boundary method with direct forcing for the simulation of particle laden flows, J. Comput. Phys. 231 (9) (2012) 3663–3684.
[22] J. Yang, F. Stern, A simple and efficient direct forcing immersed boundary framework for fluid–structure interactions, J. Comput. Phys. 231 (15) (2012) 5029–5061.
[23] A.P.S. Bhalla, R. Bale, B.E. Griffith, N.A. Patankar, A unified mathematical framework and an adaptive numerical method for fluid–structure interaction with rigid, deforming, and elastic bodies, J. Comput. Phys. 250 (2013) 446–476.
[24] J. Yang, F. Stern, A non-iterative direct forcing immersed boundary method for strongly-coupled fluid–solid interactions, J. Comput. Phys. 295 (2015) 779–804.
[25] C. Wang, J.D. Eldredge, Strongly coupled dynamics of fluids and rigid-body systems with the immersed boundary projection method, J. Comput. Phys. 295 (2015) 87–113.
[26] Y. Kim, C.S. Peskin, A penalty immersed boundary method for a rigid body in fluid, Phys. Fluids 28 (3) (2016) 033603.
[27] U. Lācis, K. Taira, S. Bagheri, A stable fluid–structure-interaction solver for low-density rigid bodies using the immersed boundary projection method, J. Comput. Phys. 305 (2016) 300–318.
[28] A.R. Koblitz, S. Lovett, N. Nikiforakis, W.D. Henshaw, Direct numerical simulation of particulate flows with an overset grid method, arXiv e-prints.
[29] F.C. Dougherty, J.-H. Kuan, Transonic Store Separation Using a Three-Dimensional Chimera Grid Scheme, Paper 89-0637, AIAA, 1989.
[30] R. Meakin, Moving Body Overset Grid Methods for Complete Aircraft Tiltrotor Simulations, Paper 93-3350, AIAA, 1993.
[31] W.D. Henshaw, D.W. Schwendeman, Moving overlapping grids with adaptive mesh refinement for high-speed reactive and non-reactive flow, J. Comput. Phys. 216 (2) (2006) 744–779.
[32] F. Zahle, J. Johansen, N.N. Sørensen, J.M.R. Graham, Wind Turbine Rotor–Tower Interaction Using an Incompressible Overset Grid Method, paper 2007-425, AIAA, 2007.
[33] W.M. Chan, Overset grid technology development at NASA Ames Research Center, Comput. Fluids 38 (3) (2009) 496–503.
[34] D.D.J. Chandar, M. Damodaran, Numerical study of the free flight characteristics of a flapping wing in low Reynolds numbers, J. Aircr. 47 (1) (2010) 141–150.
[35] A. Lani, B. Sjögreen, H.C. Yee, W.D. Henshaw, Variable high-order multiblock overlapping grid methods for mixed steady and unsteady multiscale viscous flows, part II: hypersonic nonequilibrium flows, Commun. Comput. Phys. 13 (2) (2012) 583–602.
[36] W.D. Henshaw, N.A. Petersson, A Split-Step Scheme for the Incompressible Navier–Stokes Equations, Tech. Rep., Centre for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94551, 2001.
[37] R.E. English, L. Qiu, Y. Yu, R. Fedkiw, An adaptive discretization of incompressible flow using a multitude of moving Cartesian grids, J. Comput. Phys. 254 (2013) 107–154.
[38] J.W. Banks, W.D. Henshaw, D.W. Schwendeman, Deforming composite grids for solving fluid structure problems, J. Comput. Phys. 231 (9) (2012) 3518–3547.
[39] J.W. Banks, W.D. Henshaw, B. Sjögreen, A stable FSI algorithm for light rigid bodies in compressible flow, J. Comput. Phys. 245 (2013) 399–430.
[40] J.W. Banks, W.D. Henshaw, D.W. Schwendeman, An analysis of a new stable partitioned algorithm for FSI problems, part I: incompressible flow and elastic solids, J. Comput. Phys. 269 (2014) 108–137.
[41] J.W. Banks, W.D. Henshaw, D.W. Schwendeman, An analysis of a new stable partitioned algorithm for FSI problems, part II: incompressible flow and structural shells, J. Comput. Phys. 268 (2014) 399–416.
[42] L. Li, W.D. Henshaw, J.W. Banks, D.W. Schwendeman, G.A. Main, A stable partitioned FSI algorithm for incompressible flow and deforming beams, J. Comput. Phys. 312 (2016) 272–306.
[43] J.W. Banks, W.D. Henshaw, A. Kapila, D.W. Schwendeman, An added-mass partitioned algorithm for fluid–structure interactions of compressible fluids and nonlinear solids, J. Comput. Phys. 305 (2016) 1037–1064.
[44] A. Wachs, A. Hammouti, G. Vinay, M. Rahmani, Accuracy of finite volume/staggered grid distributed Lagrange multiplier/fictitious domain simulations of particulate flows, Comput. Fluids 115 (2015) 154–172.
[45] T. Broering, Y. Lian, W. Henshaw, Numerical investigation of energy extraction in a tandem flapping wing configuration, AIAA J. 50 (11) (2012) 2295–2307.
[46] G.S. Chesshire, W.D. Henshaw, Composite overlapping meshes for the solution of partial differential equations, J. Comput. Phys. 90 (1) (1990) 1–64.
[47] W.D. Henshaw, Ogen: An Overlapping Grid Generator for Overture, Research Report UCRL-MA-132237, Lawrence Livermore National Laboratory, 1998.
[48] G.S. Chesshire, W.D. Henshaw, A scheme for conservative interpolation on overlapping grids, SIAM J. Sci. Comput. 15 (4) (1994) 819–845.
[49] H.S. Tang, S.C. Jones, F. Sotiropoulos, An overset-grid method for 3D unsteady incompressible flows, J. Comput. Phys. 191 (2003) 567–600.
[50] W.D. Henshaw, A fourth-order accurate method for the incompressible Navier–Stokes equations on overlapping grids, J. Comput. Phys. 113 (1994) 13–25.
[51] W.D. Henshaw, N.A. Petersson, A split-step scheme for the incompressible Navier–Stokes equations, in: M.M. Hafez (Ed.), Numerical Simulation of Incompressible Flows, World Scientific, 2003, pp. 108–125.
[52] A.S. Almgren, J.B. Bell, P. Colella, L.H. Howell, M.L. Welcome, A conservative adaptive projection method for the variable density incompressible Navier–Stokes equations, J. Comput. Phys. 142 (1998) 1–46.
[53] J.H. Ferziger, M. Perić, Computational Methods for Fluid Dynamics, 3rd ed., Springer, 2002.
[54] S.V. Patankar, D.B. Spalding, A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows, Int. J. Heat Mass Transf. 15 (1972) 1787–1806.
[55] N.A. Petersson, Stability of pressure boundary conditions for Stokes and Navier–Stokes equations, J. Comput. Phys. 172 (2001) 40–70.
[56] S. Balay, J. Brown, K. Buschelman, V. Eijkhout, W.D. Gropp, D. Kaushik, M.G. Knepley, L.C. McInnes, B.F. Smith, H. Zhang, PETSc Users Manual, Tech. Rep., Argonne National Laboratory, 2013.
[57] R.D. Falgout, U.M. Yang, Hypre: A Library of High Performance Preconditioners, Springer, Berlin, Heidelberg, 2002.
[58] W.D. Henshaw, On multigrid for overlapping grids, SIAM J. Sci. Comput. 26 (5) (2005) 1547–1572.
[59] U. Küttler, W.A. Wall, Fixed-point fluid–structure interaction solvers with dynamic relaxation, Comput. Mech. 43 (1) (2008) 61–72.
[60] G.K. Batchelor, An Introduction to Fluid Dynamics, Cambridge University Press, 1967.
[61] A. Nicolle, Flow Through and Around Groups of Bodies, Ph.D. Thesis, University College London, 2010.
[62] A. Nicolle, I. Eames, Numerical study of flow through and around a circular array of cylinders, J. Fluid Mech. 679 (2011) 1–31.
[63] D. Wan, S. Turek, Direct numerical simulation of particulate flow via multigrid FEM techniques and the fictitious boundary method, Int. J. Numer. Methods Fluids 51 (2006) 531–566.
[64] Y. Hu, D. Li, S. Shu, X. Niu, Modified momentum exchange method for fluid–particle interactions in the lattice Boltzmann method, Phys. Rev. E 91.
[65] A.M. Ardekani, R.H. Rangel, Numerical investigation of particle–particle and particle–wall collisions in a viscous fluid, J. Fluid Mech. 596 (2008) 437–466.
[66] G.G. Joseph, R. Zenit, M.L. Hunt, A.M. Rosenwinkel, Particle–wall collisions in a viscous fluid, J. Fluid Mech. 433 (2001) 329–346.
[67] L. Qiu, Y. Yu, R. Fedkiw, On thin gaps between rigid bodies two-way coupled to incompressible flow, J. Comput. Phys. 292 (2015) 1–29.
[68] A.F. Fortes, D.D. Joseph, T.S. Lundgren, Nonlinear mechanics of fluidization of beds of spherical particles, J. Fluid Mech. 177 (1987) 467–483.

[69] N.A. Patankar, P. Singh, D.D. Joseph, R. Glowinski, T.-W. Pan, A new formulation of the distributed Lagrange multiplier/fictitious domain method for particulate flows, Int. J. Multiph. Flow 26 (2000) 1509–1524.
[70] N.A. Patankar, A Formulation for fast Computations of Rigid Particulate Flows, Center for Turbulence Research, 2001.
[71] X.D. Niu, C. Shu, Y.T. Chew, Y. Peng, A momentum exchange-based immersed boundary–lattice Boltzmann method for simulating incompressible viscous flows, Phys. Lett. A 354 (2006) 173–182.
[72] Z. Zhang, A. Prosperetti, A method for particle simulation, J. Appl. Meth. 70 (2003) 64–74.
[73] Z.-G. Feng, E.E. Michaelides, The immersed boundary–lattice Boltzmann method for solving fluid–particles interaction problems, J. Comput. Phys. 195 (2004) 602–628.
[74] A. ten Cate, C.H. Nieuwstad, J.J. Derksen, H.E.A. Van den Akker, Particle imaging velocimetry experiments and lattice-Boltzmann simulations on a single sphere settling under gravity, Phys. Fluids 14 (2002) 4012.
[75] T. Kempe, J. Fröhlich, Collision modelling for the interface-resolved simulation of spherical particles in viscous fluids, J. Fluid Mech. 709 (2012) 445–489.
[76] A. Wachs, A DEM-DLM/FD method for direct numerical simulation of particulate flows: sedimentation of polygonal isometric particles in a Newtonian fluid with collisions, Comput. Fluids 38 (2009) 1608–1628.